

TUSB9260/TUSB9261 Flash Burner

User's Guide



Literature Number: SLLU125B
August 2010—Revised August 2011

1	Introduction	7
1.1	Reference Material	7
1.2	Definitions	7
1.3	Required Equipment	8
2	Installing the Flash Burner Software	9
2.1	Running the Setup Program	9
2.2	Connecting the TUSB9260/TUSB9261 Based HW	11
3	Flash Burner GUI description	17
3.1	HID Interface	17
3.2	SCSI Interface	22
4	Using the Flash Burner Software	25
4.1	Opening the Flash Burner Software	25
4.2	Flash Burner GUI options	26
4.3	Editing USB Descriptors	28
4.4	Selecting a Compatible Device	29
4.5	Selecting a Compatible Firmware Binary File	30
4.6	Burning a FW Binary File Into the SPI Flash	31
4.7	Erasing FW in the SPI Flash	32
4.8	Exporting the EEPROM Data to a File	33
5	Troubleshooting	34
5.1	5.1 Re-Installing Flash Burner Driver Instance Manually	34
5.2	TUSB9260/TUSB9261 Not Detected Due to a Corrupted FW in the SPI Flash	38
5.3	TUSB9260/TUSB9261 Unrecognized By the Flash Burner GUI and Device Manager	38
5.4	GUI Succeeded, But Device is Not Properly Programmed	39

List of Figures

2-1.	.NET Framework 3.5 Installation	9
2-2.	Driver Installation Warning on XP and Vista/Win7	10
2-3.	Security Window for Vista/Win7	10
2-4.	Software Installation Completed	11
2-5.	TUSB9260/TUSB9261 Instances With Blank or Not Connected SPI Flash	12
2-6.	TUSB9260/TUSB9261 Instances With FW Containing HID Interface	13
2-7.	TUSB9260/TUSB9261 Instances With FW Containing Mass Storage Instance Only (No HID)	14
2-8.	Instance Properties When Blank or Not Connected SPI Flash	15
2-9.	Instance Properties With FW Containing HID Interface	15
2-10.	Instance Properties With FW Containing Mass Storage Instance Only (No HID)	16
3-1.	Descriptors and FW Location	17
3-2.	Get HID Devices Flow Diagram	20
3-3.	GUI Burner/Re-Programmer Flow Diagram	21
3-4.	Listing Compatible Mass Storage Devices	23
3-5.	Erasing the Device's Flash Through SCSI Commands	24
4-1.	Flash Burner Software Locations	25
4-2.	Flash Burner GUI	26
4-3.	Flash Burner Software Version	26
4-4.	Selecting Options on the Flash Burner GUI	27
4-5.	Getting Current Serial Number	28
4-6.	Descriptors Info Group-Box	28
4-7.	Editing USB Descriptors	29
4-8.	List for Selecting a Compatible TUSB9260/TUSB9261 Device	30
4-9.	Selecting a Valid FW File	30
4-10.	Burning a Valid FW File into the SPI Flash	31
4-11.	SPI Flash Programming Succeeded	31
4-12.	Operating System Asking for a Reboot	31
4-13.	Erasing SPI Flash With HID FW Already Burned	32
4-14.	Erase Flash Command Succeeded	32
4-15.	Erasing the Device's Flash from USB Mass Storage Instance	33
4-16.	Export Button	33
4-17.	Saving the EEPROM	33
5-1.	TUSB9260/TUSB9261 Boot Loader Instance Not Properly Recognized	35
5-2.	Manual Installation for a TUSB9260/TUSB9261 HW Instance	36
5-3.	Manual Installation Wizard to Avoid Windows Update Online	37
5-4.	Manual Installation Wizard to Perform an Automatic Search	38
5-5.	Data Transmission During Flash Programming	39

List of Tables

3-1.	HID Reports	19
3-2.	SCSI Commands	22
5-1.	OP-Codes	39

Introduction

This document will serve as a user's manual for the TUSB9260/TUSB9261's GUI Flash Burner/Re-programmer. It provides details on using the Flash Burner software used with TUSB9260/TUSB9261 based devices. The software is a Windows-based tool allowing in-circuit programming of the SPI Flash via the Universal Serial Bus (USB). A valid Firmware file must be used in order to properly use the TUSB9260/TUSB9261 device as a USB 3.0 to SATA bridge mode.

The TUSB9260/TUSB9261 Flash Burner/Re-programmer is a Windows based application allowing the SPI Flash to be programmed via USB with TUSB9260/TUSB9261 based boards.

The term 'Flash Burner' is equivalent to 'TUSB9260/TUSB9261 Flash Burner/Re-programmer.'

1.1 Reference Material

All material referenced by this document have been placed on the TUSB9260/TUSB9261 Program Management/Design SharePoint sites if not available from any external public source.

1.2 Definitions

The following are useful definitions for your reference:

- GUI – Graphical User Interface
- HID – Human Interface Device
- SPI – Serial Peripheral Interface
- POR – Power-On Reset
- FW – Firmware
- IO – Input/Output
- HAL – Hardware Access Layer
- UAS – USB Attached SCSI
- BOT – Bulk Only Transfer
- BL – Boot Loader
- SW – Software
- MS – Microsoft
- DMA – Direct Memory Access
- VID – Vendor ID
- PID – Product ID
- SCSI – Small Computer System Interface

1.3 Required Equipment

The following is required in order to use the EEPROM programmer software:

- TUSB9260/TUSB9261 EVM board
- SPI Flash device (see Appendix A for acceptable devices)
- USB interface Cable (Type A connector to Type B connector)
- PC running Windows XP™, Windows Vista™ or Windows 7™ (32-bit and 64-bit OS)
- USB 2.0 or 3.0 host already present in your system
- TUSB9260/TUSB9261 Firmware version 0.95 or greater

Installing the Flash Burner Software

2.1 Running the Setup Program

Extract the setup program on a directory of your choice. Run the 'setup.exe' file and let the 'Installshield Wizard' guide you through the pre-installer installation process.

The Flash Burner program needs MS '.NET Framework 3.5' as a pre-requisite in order to work properly. The installer will guide you through this process in case there's no such .NET framework version installed on your machine. Ensure that you have a proper internet connection since the installation will be performed through a web download directly from the MS site (see [Figure 2-1](#)).

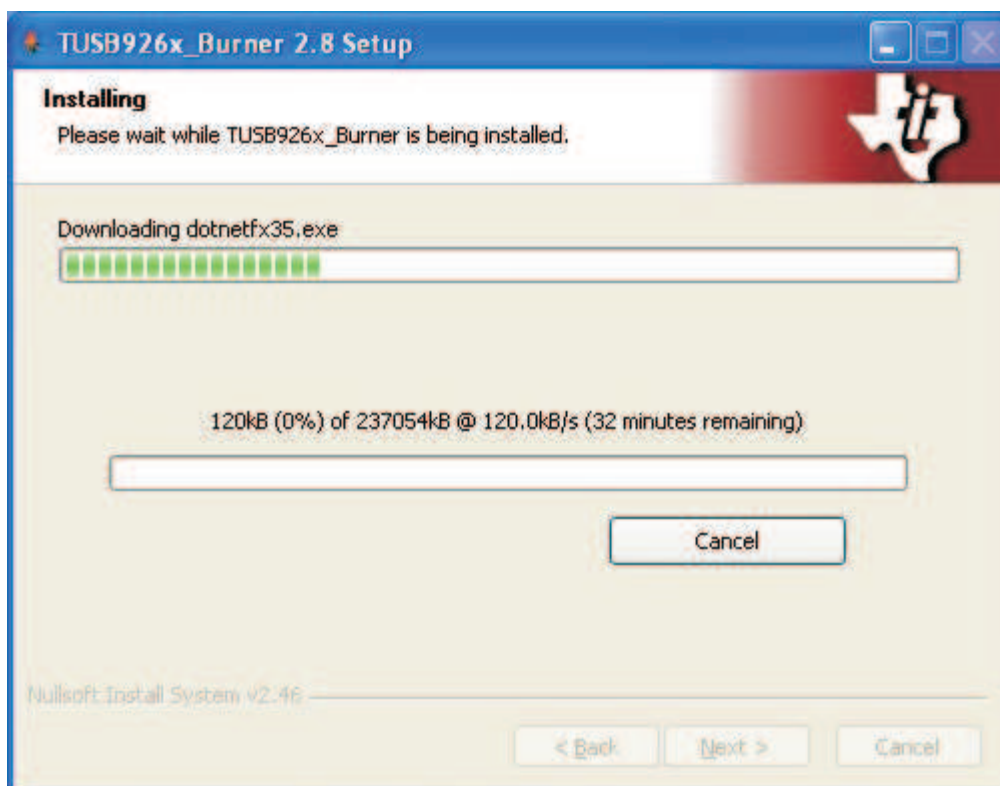


Figure 2-1. .NET Framework 3.5 Installation

After the .NET validation has finished, the installer will copy the necessary files to your local disk and will execute a driver co-installer which will install the "FlashBurner" driver in your system. Depending on your system settings, you may get a warning message or security window during the driver installation process. When prompted, accept the driver installation as shown in [Figure 2-2](#) and [Figure 2-3](#).

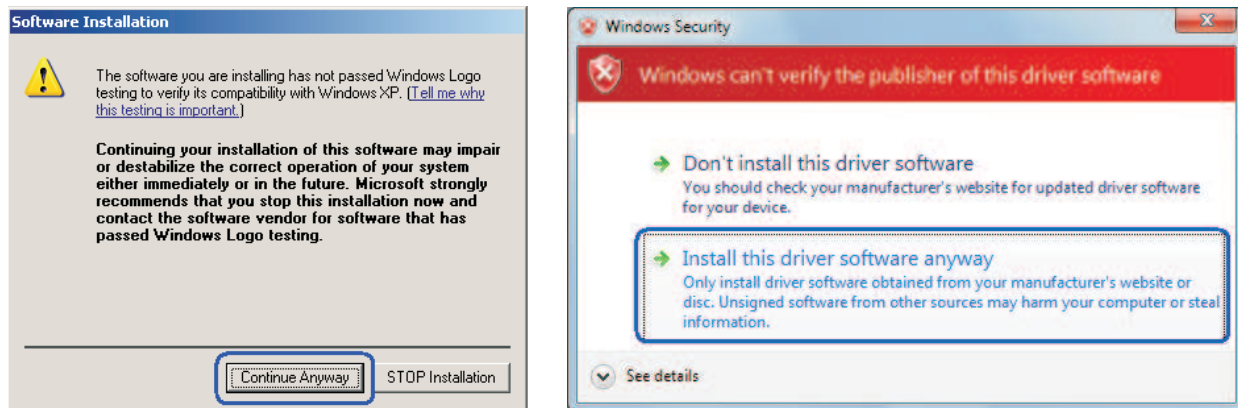


Figure 2-2. Driver Installation Warning on XP and Vista/Win7

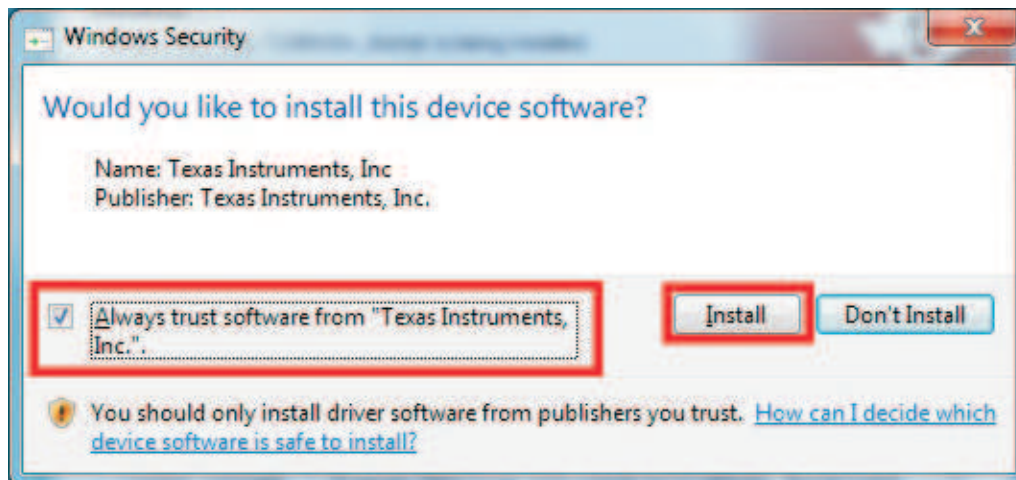


Figure 2-3. Security Window for Vista/Win7

After all the necessary files have been copied into your system, the installer will give you the option to “Restart” the system in order to properly update the files.

It is highly recommended to choose “Reboot now” and click “Finish” when the installer wizard indicates that the Flash Burner software installation has been completed (see [Figure 2-4](#)).

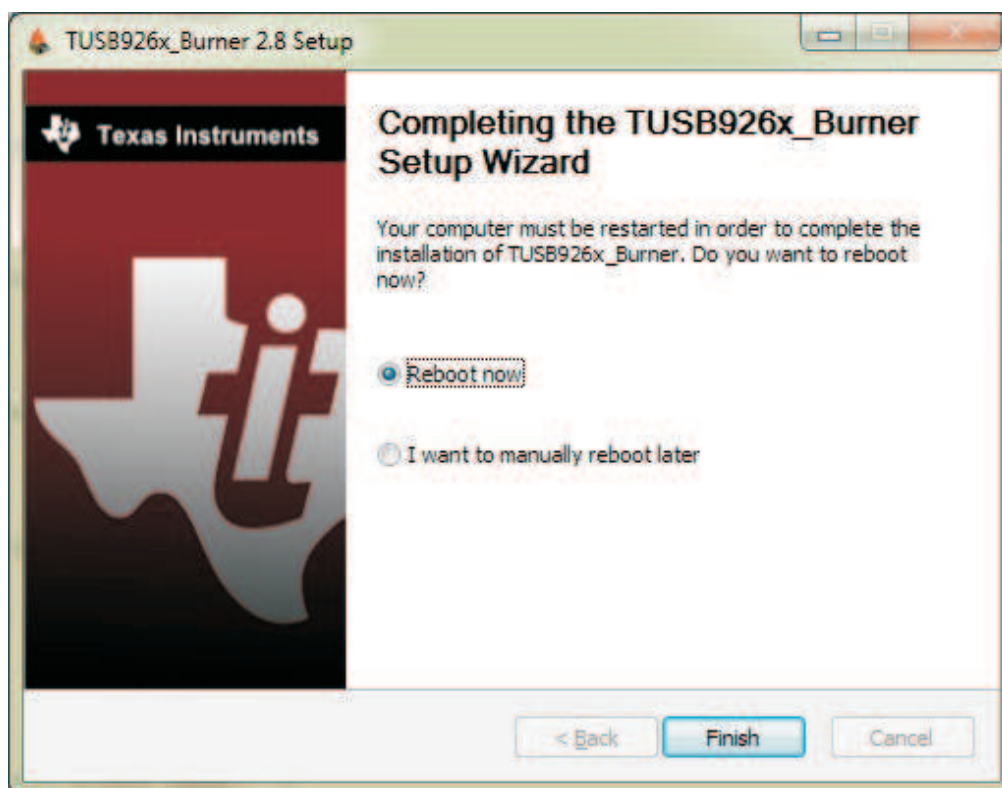


Figure 2-4. Software Installation Completed

2.2 Connecting the TUSB9260/TUSB9261 Based HW

You can now connect your TUSB9260/TUSB9261 EVM board to any USB port available on your PC.

There are three different cases of how your TUSB9260/TUSB9261 EVM board can be enumerated by the OS and it will depend on the content of the device's SPI Flash:

1. Blank or not connected SPI Flash (see [Figure 2-5](#)).
2. Previously burned FW in the SPI Flash containing TI's predetermined HID instances and functions (see [Figure 2-6](#)).
3. Previously burned FW in the SPI Flash containing a mass storage instance only (no HID) (see [Figure 2-7](#)).

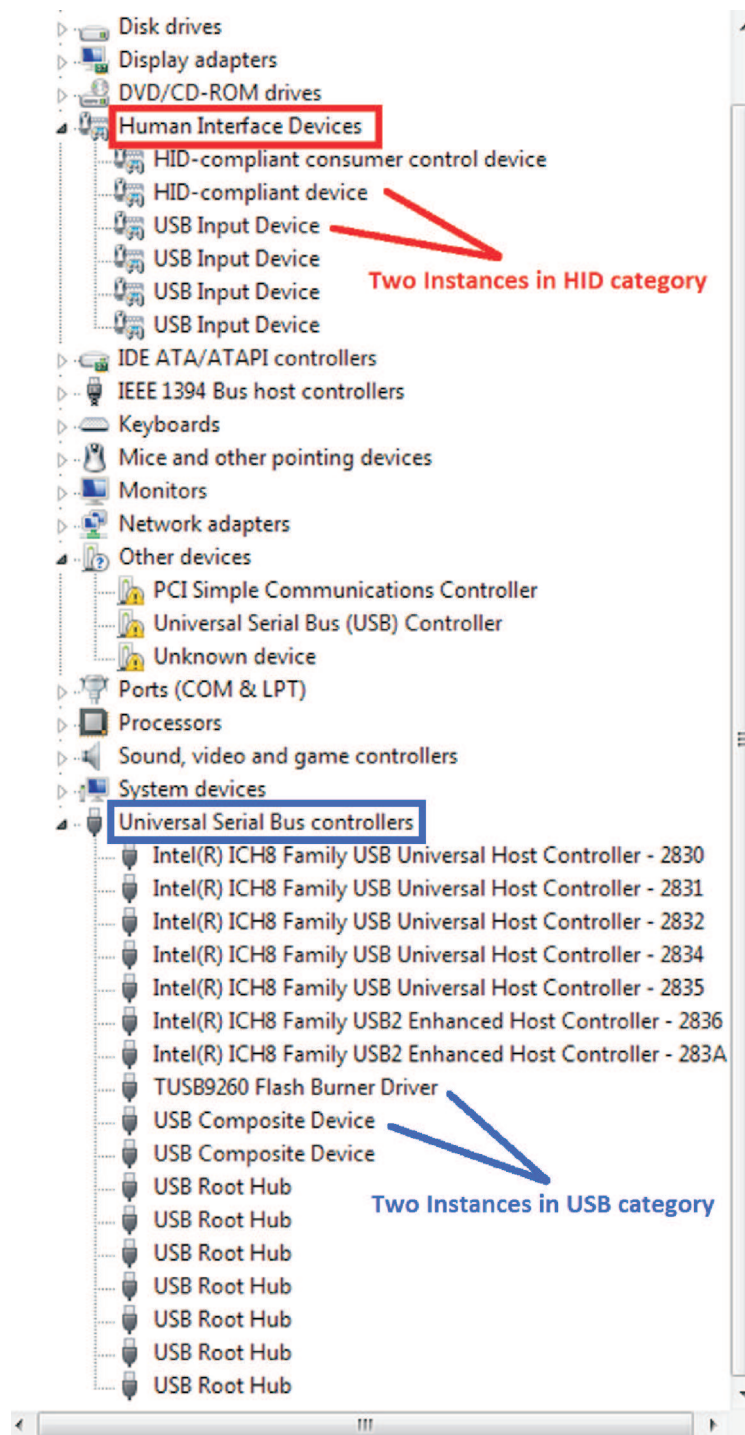


Figure 2-5. TUSB9260/TUSB9261 Instances With Blank or Not Connected SPI Flash

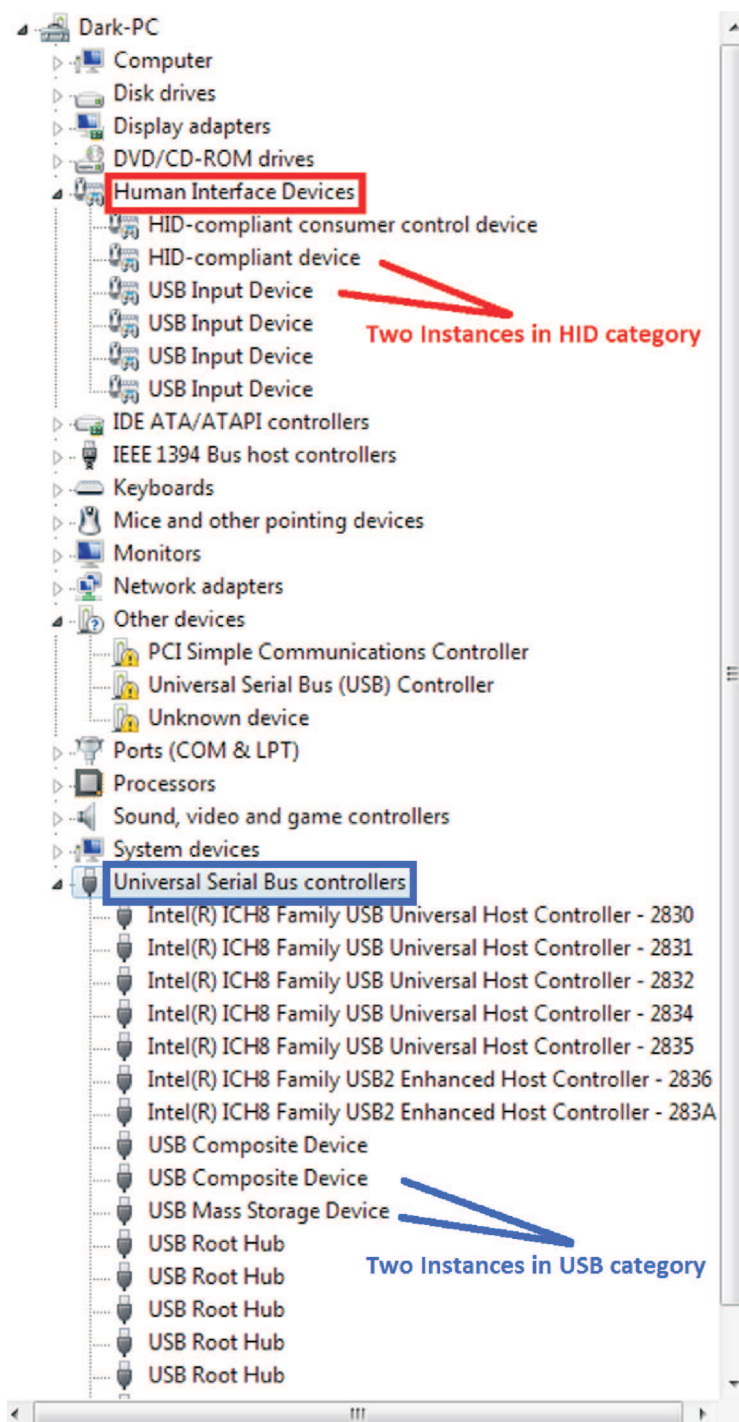


Figure 2-6. TUSB9260/TUSB9261 Instances With FW Containing HID Interface

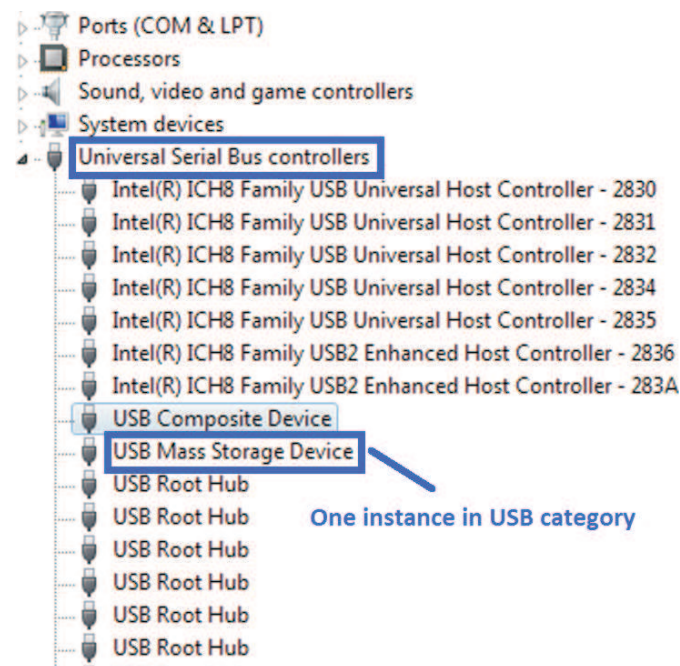


Figure 2-7. TUSB9260/TUSB9261 Instances With FW Containing Mass Storage Instance Only (No HID)

After the TUSB9260/TUSB9261 EVM board has been properly enumerated, you can double-click on each new instance in order to see the properties so you'll be able to check that all of them are related to the TUSB9260/TUSB9261 device. VID, PID and descriptors are displayed with TI's default information.

It's important to note that depending of the content present in the SPI flash the information for each instance may vary.

The following are the three cases you may find while looking into your TUSB9260/TUSB9261 EVM board instance properties:

1. Instance properties for a TUSB9260/TUSB9261 EVM board with blank, previously erased or not connected SPI Flash (see [Figure 2-8](#)).
2. Instance properties for a TUSB9260/TUSB9261 EVM board with previously burned FW in the SPI Flash containing TI's predetermined HID instances and functions (see [Figure 2-9](#)).
3. Instance properties for a TUSB9260/TUSB9261 EVM board with previously burned FW in the SPI Flash containing a mass storage instance only (no HID) (see [Figure 2-10](#)).

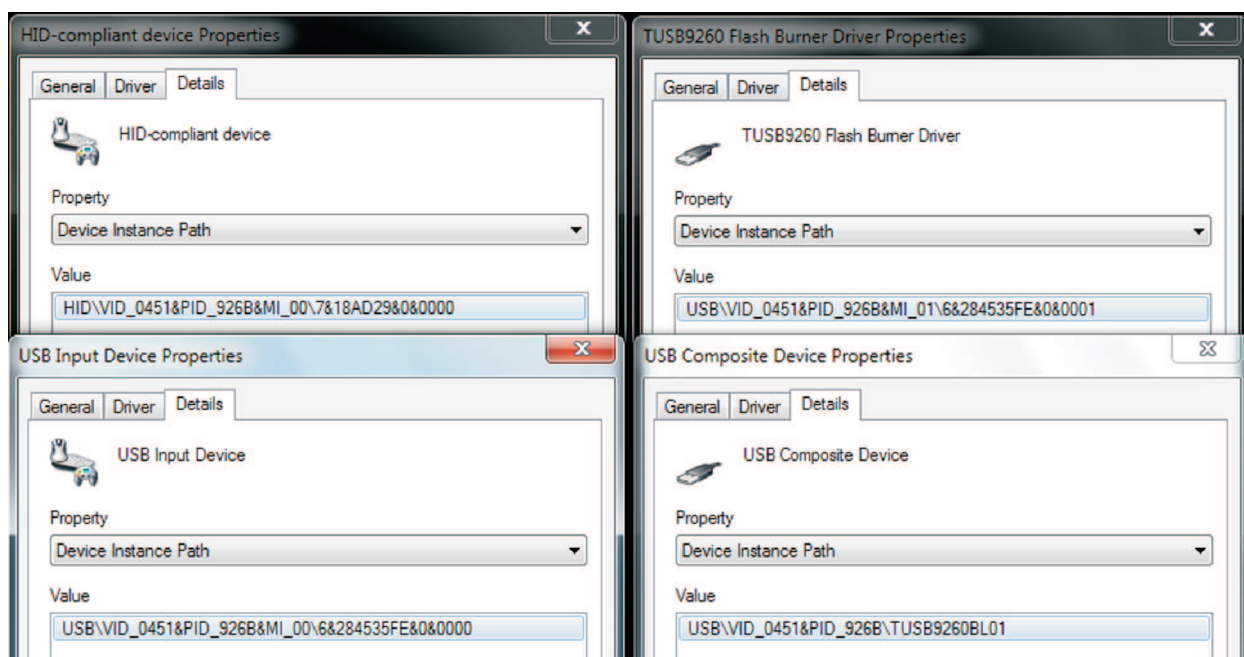


Figure 2-8. Instance Properties When Blank or Not Connected SPI Flash

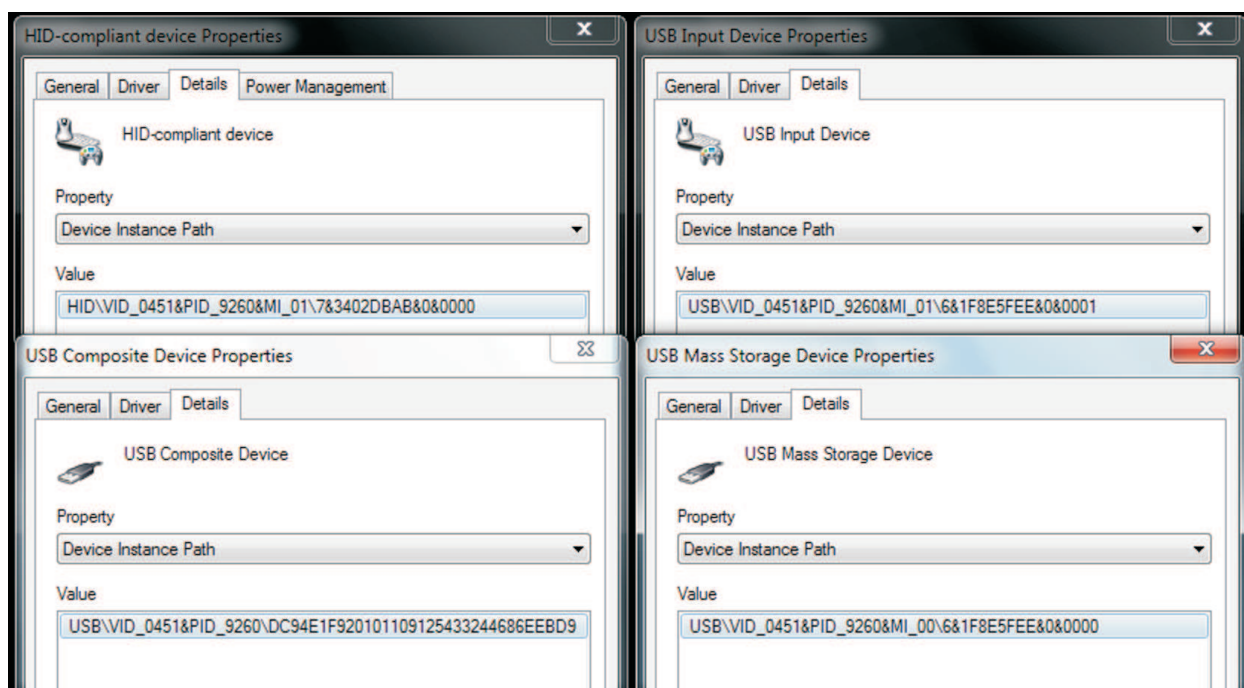


Figure 2-9. Instance Properties With FW Containing HID Interface

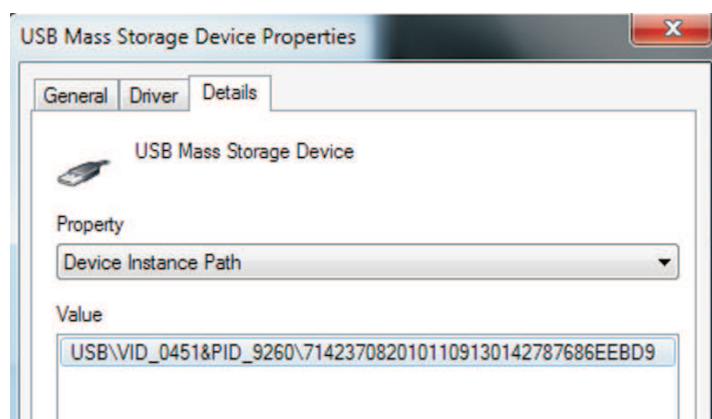


Figure 2-10. Instance Properties With FW Containing Mass Storage Instance Only (No HID)

Flash Burner GUI description

3.1 HID Interface

The TUSB9260/TUSB9261's GUI Flash Burner/Re-programmer is a user-friendly application to program the SPI Flash and allow the Apps to "Peek and Poke" registers over USB. The Flash Burner GUI interacts directly with the TUSB9260/TUSB9261 device using a set of seven HID Reports and bulk transfers to the device's endpoint 2. Each one of the HID reports will instruct the "Boot Loader"/"Firmware" in the device to perform a specific task.

- **USB_HID_SETUP_DOWNLOAD_DATA**

This report is used to prepare the device for a data download. After issuing this HID call, the data will be sent to the device through a bulk transfer to the device's endpoint 2. The Flash Burner GUI achieves this through an IOCTL call using the TUSB9260/TUSB9261 Flash Burner driver, which is installed along with the Flash Burner utility. The data contains the device descriptors and Firmware formatted as represented in [Figure 3-1](#).

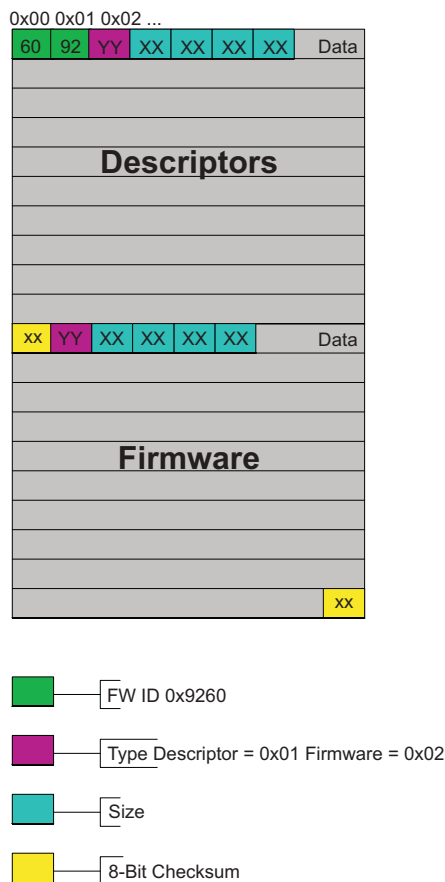


Figure 3-1. Descriptors and FW Location

Through this report, the user can specify whether the data will be downloaded to the device's RAM or SPI Flash. Only the Boot Loader device supports this HID report.

- **USB_HID_RESET_FLASH_BURNER_DEVICE**
This report will instruct the device to perform a soft reset. Both the Boot Loader and the Firmware support this HID report.
- **USB_HID_POISON_FLASH**
This report will instruct the Boot Loader to erase the content of the SPI Flash. Only the Firmware supports the HID report.
- **USB_HID_READ_REG**
This report will read the content of a specific memory address. Only the Firmware supports the HID report.
- **USB_HID_WRITE_REG**
This report will set the content of a specific memory address with the specified data. Only the Firmware supports the HID report.
- **USB_HID_ENABLE_REPROGRAM**
This report is used to prepare a device that already has a FW on it, to be re-programmed. Only the Firmware supports the HID report.
- **USB_HID_IS_FLASH_PRESENT**
This report will help us to determine whether the device has an SPI Flash or not. Only the Boot Loader supports this HID call.
- **USB_HID_GET_FIRMWARE_VERSION**
Through this report we can determine the current firmware version operating in our device. Obviously, only the Firmware supports this HID report.

[Table 3-1](#) describes the data contained on these reports.

Table 3-1. HID Reports

HID REPORT	BYTE 0	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5	BYTE 6	BYTE 7	BYTE 8
USB_HID_SETUP_DOWNLOAD_DATA	Opcode = 0x01	Valid Values: 0x00 or 0x01. 0x00 = Flash 0x01 = RAM	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Size [D7 - D0]	Size [D15 - D8]	Size [D23 - D16]	Size [D31 - D24]
USB_HID_RESET_FLASH_BURNER_DEVICE	Opcode = 0x02	Valid Values: 0x00 or 0x01. 0x00 = Flash 0x01 = RAM	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00
USB_HID_POISON_FLASH	Opcode = 0x03	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00
USB_HID_READ_REG	Opcode = 0x04	Address [A7 - A0]	Address [A15 - A8]	Address [A23 - A16]	Address [A31 - A24]	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00
USB_HID_WRITE_REG	Opcode = 0x05	Address [A7 - A0]	Address [A15 - A8]	Address [A23 - A16]	Address [A31 - A24]	Data [D7 - D0]	Data [D15 - D8]	Data [D23 - D16]	Data [D31 - D24]
USB_HID_ENABLE_REPROGRAM	Opcode = 0x07	Valid Values: 0x00 or 0x01. 0x00 = Re-Programming Disabled 0x01 = Re-Programming Enabled	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00
USB_HID_IS_FLASH_PRESENT	Opcode = 0x08	Valid Values: 0x00 or 0x01. 0x00 = Flash 0x01 = No Flash	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00
USB_HID_GET_FIRMWARE_VERSION	Opcode = 0x09	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Not used Value should be set to 0x00	Firmware Minor Version	Firmware Major Version

In order to recognize an HID device as 9260 compatible, the device must respond with a 0x9260 to an HID Feature report request. [Figure 3-2](#) describes this process in detail.

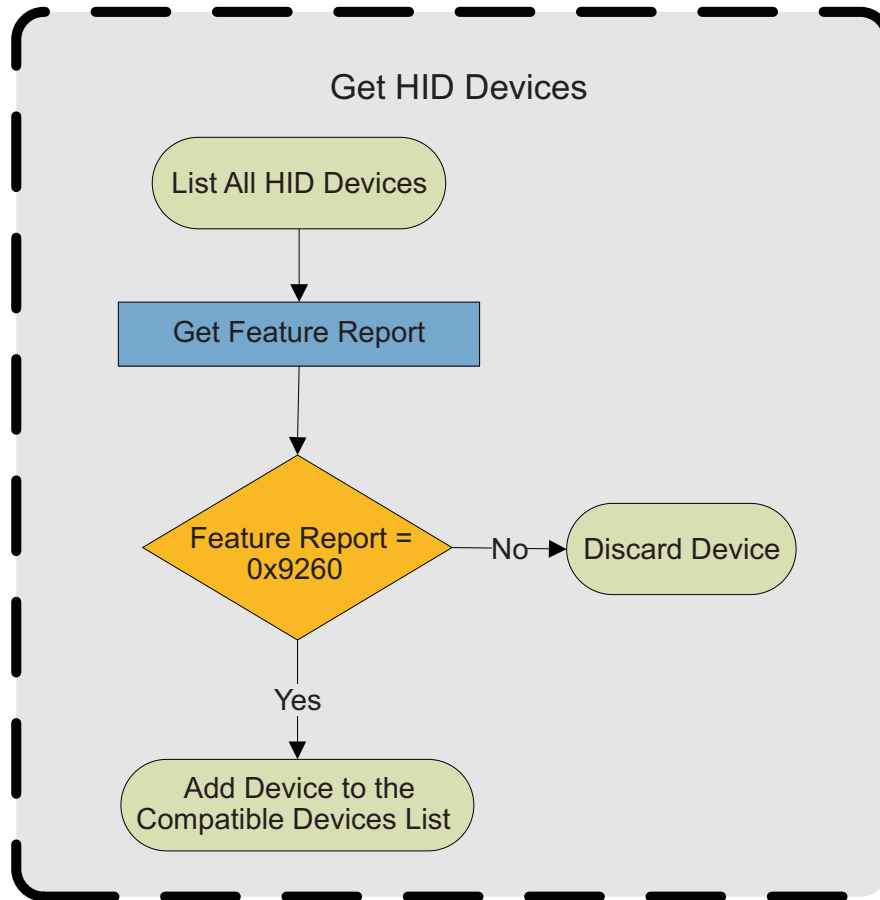


Figure 3-2. Get HID Devices Flow Diagram

After the GUI has successfully detected at least one 9260 HID compatible device, we will need to determine whether the device has already been programmed or not. We can achieve this by getting the device's VID and PID.

In case we have a device with a VID = 0x0451 and a PID = 0x926B, we can assume the device is NOT programmed and we will be ready to do so. Otherwise, we will need to take the device to its original configuration (by erasing/poisoning the device's SPI Flash) before sending the new data to be programmed.

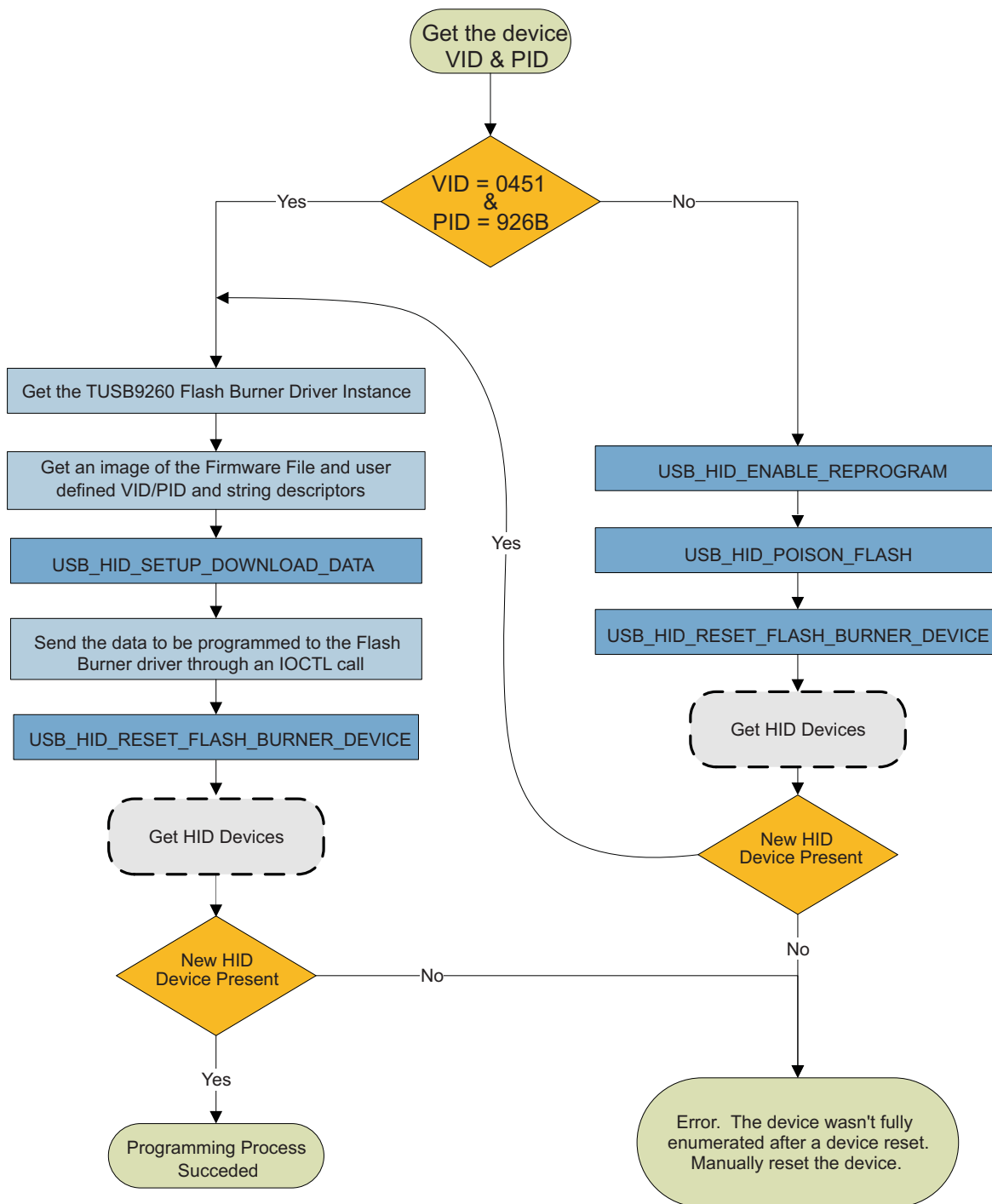


Figure 3-3. GUI Burner/Re-Programmer Flow Diagram

3.2 SCSI Interface

When having a previously burned Firmware in our device's SPI Flash containing a Mass Storage only interface, we will require additional means to erase our device's Flash when updating the Firmware content. That's why (Starting on Firmware version 0.95) a set of TI (Vendor) specific SCSI commands have been implemented, providing us the necessary means to communicate specific commands to our device whenever the HID interface is missing. The currently implanted SCSI commands are:

- **SCSI_TI_FLASH_UNLOCK**

Similarly to the USB_HID_ENABLE_REPROGRAM, this command prepares the device to be re-programmed. This command serves as a lock to ensure no accidental flash erasing is performed.

- **SCSI_TI_FLASH_ERASE**

As with the USB_HID_POISON_FLASH report, this command instructs the device to erase the content of the SPI Flash.

- **SCSI_TI_GET_PID**

This SCSI command will always return a constant data (0x9260) that will help us to identify our device among other storage devices in the system, similarly to the HID's feature report implementation

- **SCSI_TI_GET_FW_VERSION**

Through this command we can determine the current firmware version operating in our device.

- **SCSI_TI_GET_USB_SPEED**

Through this command we get an indicator representing the current USB connection speed.

- **SCSI_TI_DEVICE_RESET**

Similarly to the USB_HID_RESET_FLASH_BURNER_DEVICE report, this command instructs the device to perform a soft reset. This command is specially required after erasing the device's flash, so its BootLoader instance can come-up again.

Table 3-2 describes the data contained on the above detailed commands.

Table 3-2. SCSI Commands

SCSI COMMAND	COMMAND DATA	RETURN DATA
SCSI_TI_FLASH_UNLOCK	0xE1	No Data Return
SCSI_TI_FLASH_ERASE	0xE2	No Data Return
SCSI_TI_GET_PID	0xE3	2 Bytes – 0x9260
SCSI_TI_GET_FW_VERSION	0xE4	2 Bytes Byte 0 – FW Minor Version Byte 1 – FW Mayor Version
SCSI_TI_GET_USB_SPEED	0xE5	1 Byte USB_LOW_SPEED = 0, USB_FULL_SPEED = 1, USB_HIGH_SPEED = 2, USB_SUPER_SPEED = 3, USB_SPEED_UNKNOWN = 4
SCSI_TI_DEVICE_RESET	0xE6	No Data Return

Figure 3-4 describes the TI (Vendor) specific SCSI commands usage while listing compatible TUSB9260/TUSB9261 based disks over the Flash Burner GUI.

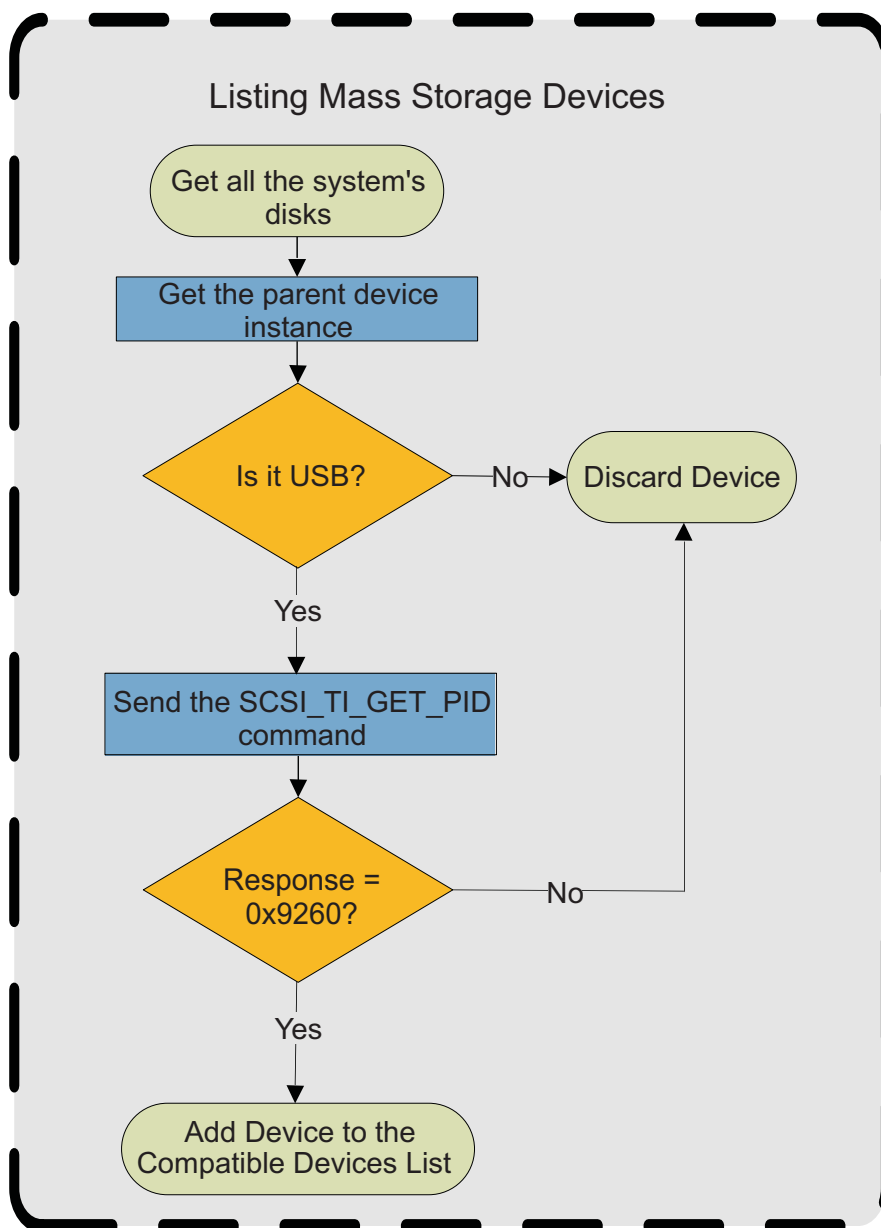


Figure 3-4. Listing Compatible Mass Storage Devices

During the flash erasing process, TI (Vendor) specific SCSI commands are issued in the sequence described in [Figure 3-5](#).

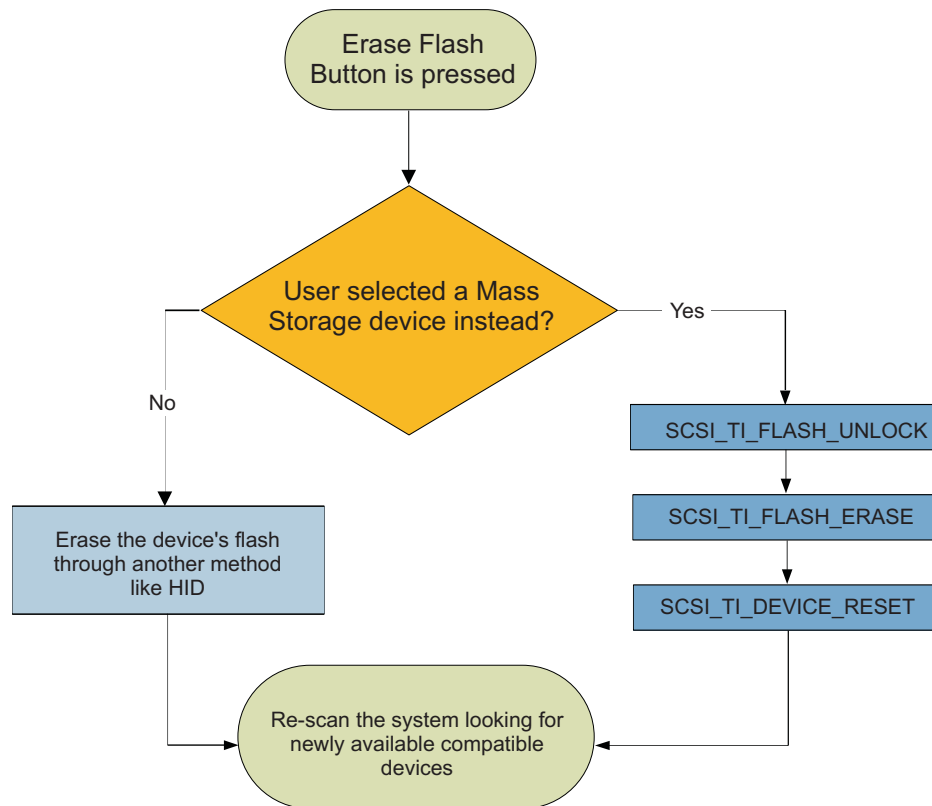


Figure 3-5. Erasing the Device's Flash Through SCSI Commands

Using the Flash Burner Software

4.1 Opening the Flash Burner Software

After you've already checked that no missing instances of the TUSB9260/TUSB9261 EVM are present at device manager, you can access the Flash Burner utility by clicking on the "TUSB9260/TUSB9261 Flash Burner" shortcut added on your desktop or by going to "Start → Texas Instruments Inc → TUSB9260/TUSB9261_FlashBurner → TUSB9260/TUSB9261 Flash Burner" (see [Figure 4-1](#)).

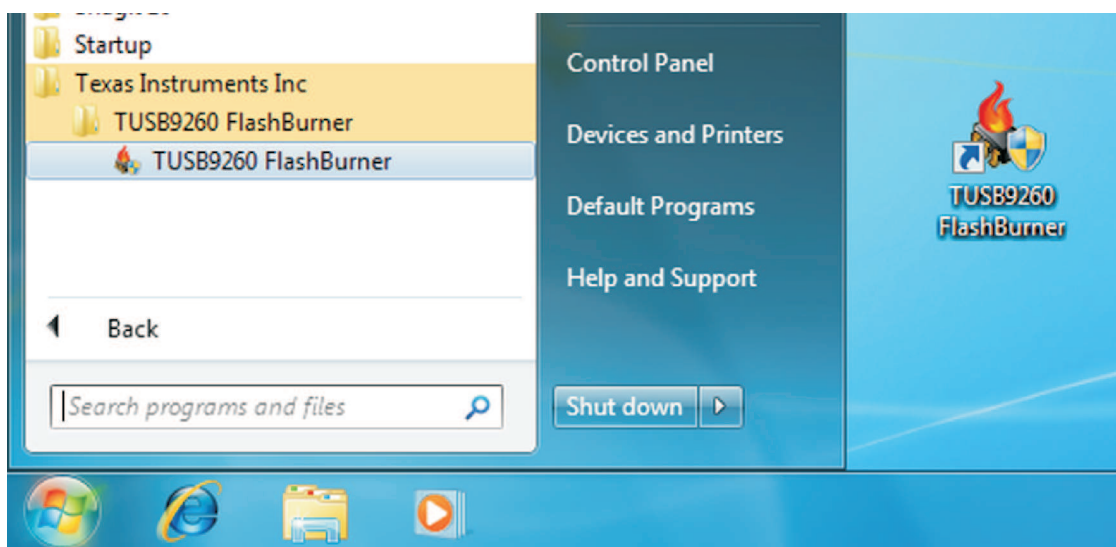


Figure 4-1. Flash Burner Software Locations

NOTE: Administrator rights are required under Windows Vista and Windows 7 to execute this application.

After executing the TUSB9260/TUSB9261 Flash Burner application, the following user interface will show up:

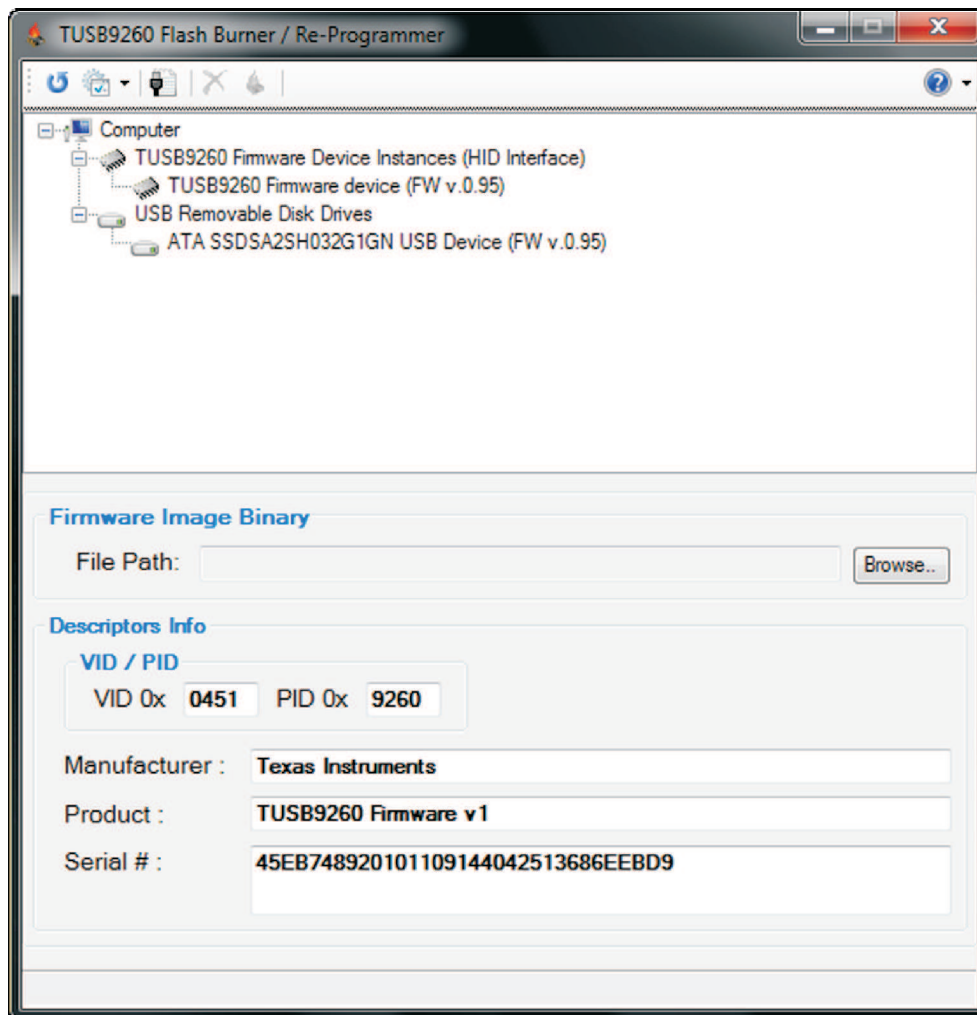


Figure 4-2. Flash Burner GUI

In case you want to check for the GUI version you are using, click on “Help” drop down menu and choose “About” in order to display the application information (see [Figure 4-3](#)).

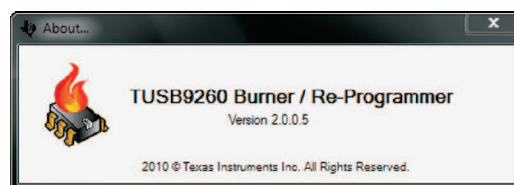


Figure 4-3. Flash Burner Software Version

4.2 Flash Burner GUI options

The “Options” menu enables the user to change different aspects of the Flash Burner GUI configuration. In order to access the options menu items, simply click on the “Options” button (see [Figure 4-4](#)).

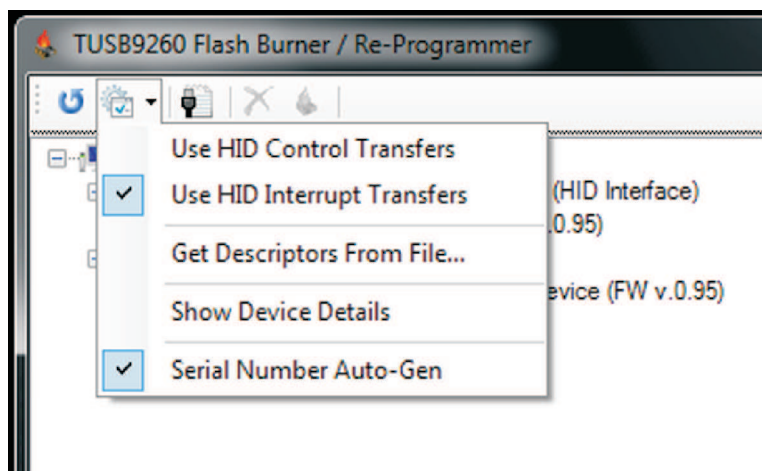


Figure 4-4. Selecting Options on the Flash Burner GUI

The following configuration aspects can be changed from the “Options” menu:

- (a) **HID transfer type:** User’s can select between using either “Control Transfers” or “Interrupt Transfers” (default).

NOTE: If for some reason you need to use “HID Control transfers”, make sure you are at least using a PG2.5 device and FW v0.83; otherwise, you will get an error message and your device might become unresponsive until resetting the device.

- (b) **Get Descriptors from File:** By setting this option, the user can get a set of descriptor’s settings from a descriptors file (*.desc) previously generated with the “Advance Descriptors Editor” (refer to [Section 4.3](#) for more details).

NOTE: By un-checking this option the application will use default descriptor’s settings.

- (c) **Show Device Details:** By setting this option, the application will show a panel displaying additional information about the selected device.
- (d) **Serial Number Auto-Gen:** By setting this option, the Flash Burner GUI will automatically generate a unique serial number for every programmed device.
The auto-generated serial number is made of:
- Random number (8 chars)
 - Date and time (Year - 4 chars, Month – 2 chars, Day – 2 chars, Hour – 2 chars, Minutes – 2 chars, Seconds – 2 chars, Milliseconds – 3 chars)
 - Serial number of the HDD running Windows (8 chars)

Additionally to the serial number auto-generation feature, users can also get the serial number from its current device just in case they need to keep it from upcoming FW updates.

In order to achieve this, after selecting a compatible device from the Flash Burner GUI device’s list, the “Get Current Serial Number” button will be displayed within the “Descriptors Info” group-box. By clicking this button (see [Figure 4-5](#)) the Flash Burner GUI will get the current serial number on your device and will automatically disable the serial number auto-generation feature if it’s enabled.

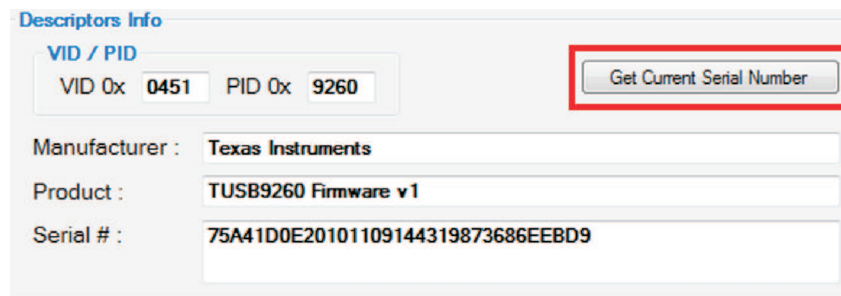


Figure 4-5. Getting Current Serial Number

4.3 Editing USB Descriptors

USB descriptors provide the host with all the necessary information to describe your USB device, so it's very important that any change to these values is carefully done.

Basic descriptor's information can be changed from the TUSB9260/TUSB9261 Flash Burner GUI main's form. Within the "Descriptors Info" group-box there is a series of text box controls (see [Figure 4-6](#)) that will let you enter customized information about your device such as:

- **VID** – Vendor ID. 4 characters long (assigned by USB Org)
- **PID** – Product ID. 4 characters long (assigned by the manufacturer)
- **Manufacturer String Descriptor** – Maximum 30 characters long
- **Product String Descriptor** – Maximum 30 characters long
- **Serial Number String Descriptor** – Must be unique for each device. Maximum 64 characters long.

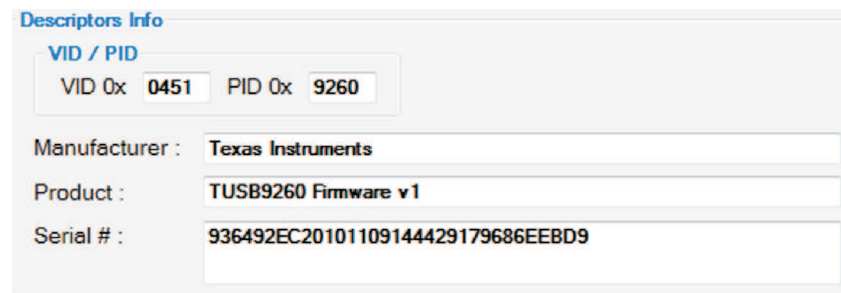


Figure 4-6. Descriptors Info Group-Box

The Flash Burner GUI also provides the means to edit additional descriptor's information through the "Advanced Descriptor's Editor" tool.

To open the "Advanced Descriptors editor" tool interface, click on the editor's button located on the tool bar menu at the top of the Flash Burner GUI.

Once the descriptor's editor is open (see [Figure 4-7](#)), the user will be able to check the value of each individual descriptor to be used, edit some additional descriptor's data and enable/disable some of the interfaces used by the device.

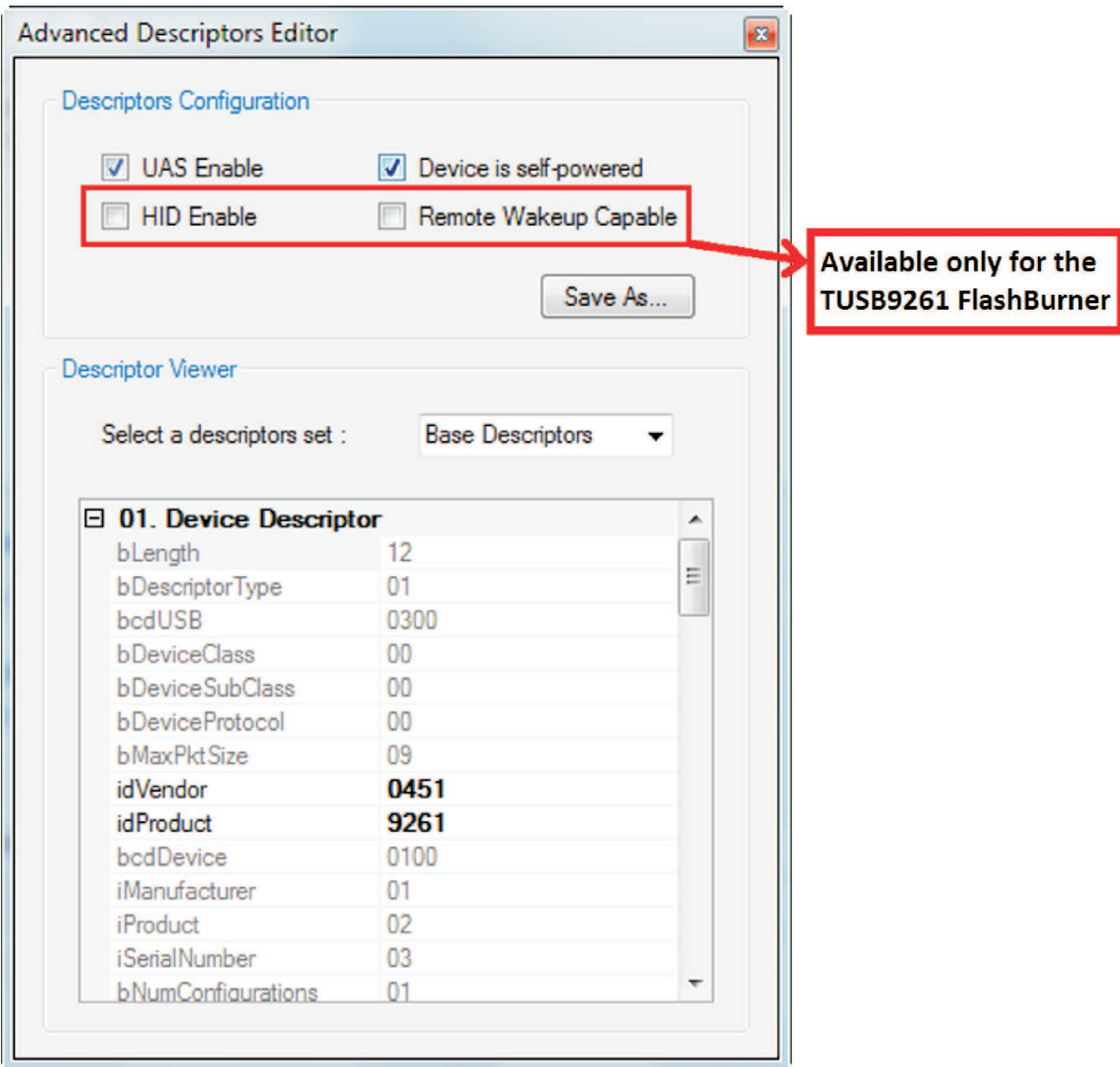


Figure 4-7. Editing USB Descriptors

Through this tool, users can also save their descriptor's configuration to a descriptors (*.desc) file for future use. That file can later be loaded from the "Options" menu (refer to [Section 4.2](#)).

To save your current descriptors to a file, simply click on the "Save As..." button, select a file name and location from the resulting dialog and click "OK".

NOTE: Make sure that all the enabled interfaces are supported by the FW that will be burned in the SPI Flash.

4.4 Selecting a Compatible Device

The Flash Burner GUI will automatically identify all the compatible devices already connected on your system and they will be listed on the top of your Flash Burner application instance. Through this list, you can choose among all the TUSB9260/TUSB9261 based devices detected (see [Figure 4-8](#)). For this example, we connected a TUSB9260/TUSB9261 EVM board with a valid FW image. Depending on the device you select, the toolbar buttons will be enabled so you can perform any of the available tasks.



Figure 4-8. List for Selecting a Compatible TUSB9260/TUSB9261 Device

4.5 Selecting a Compatible Firmware Binary File

In the middle of the Flash Burner GUI, there is a group-box identified as “Firmware Image Binary” and this has a browse button that will let you choose the *.BIN file to be burned into the SPI Flash. Click on this “Browse” button and select the appropriate FW file located in your system. Click on “Open” afterwards (see [Figure 4-9](#)).

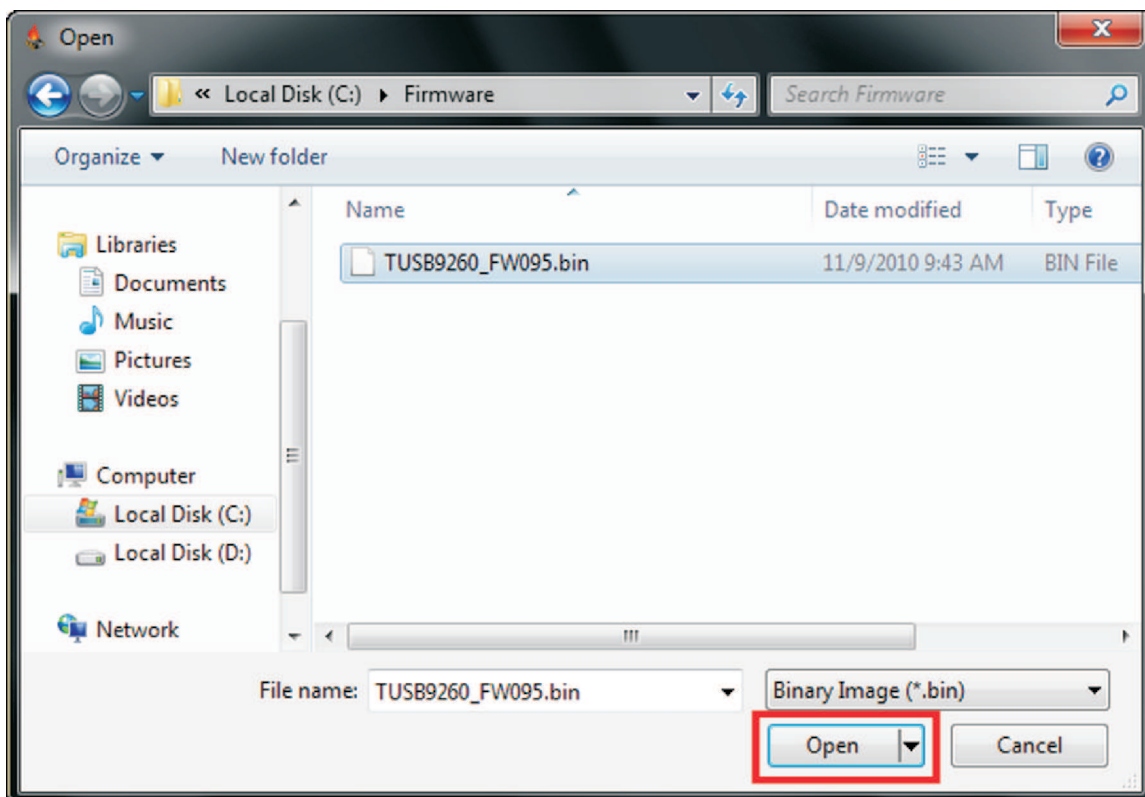


Figure 4-9. Selecting a Valid FW File

4.6 Burning a FW Binary File Into the SPI Flash

After selecting a valid FW file, you can now click on the “Program” button (see [Figure 4-10](#)).

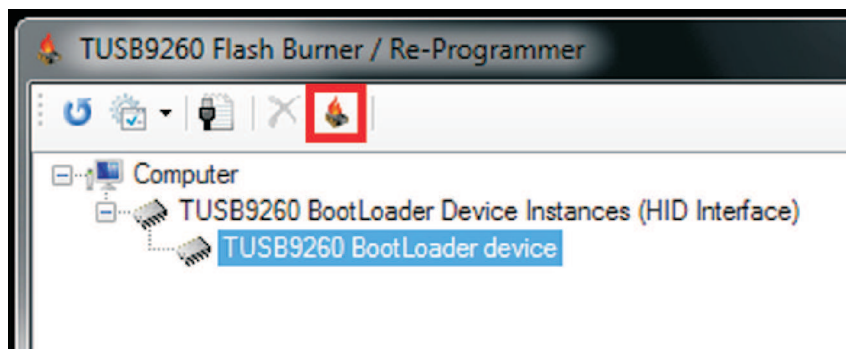


Figure 4-10. Burning a Valid FW File into the SPI Flash

When the FW file has been properly burned into the SPI Flash, a message box showing “Flash Programming Succeeded” will show up. Click on “OK” to continue (see [Figure 4-11](#)).

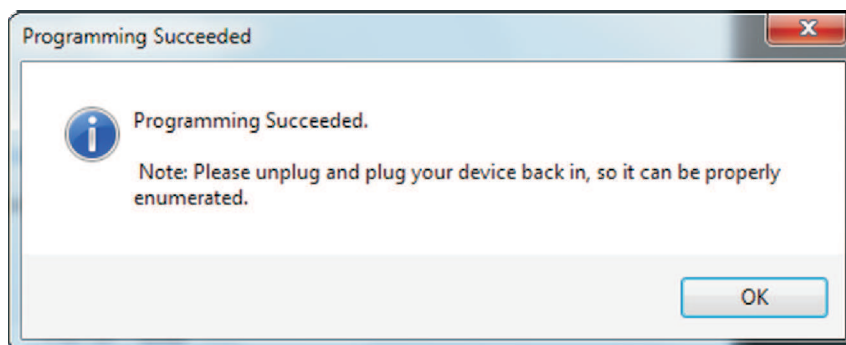


Figure 4-11. SPI Flash Programming Succeeded

A reboot is recommended after successfully programming a FW file into the SPI Flash device. Although it is not mandatory, the operating system will sometimes ask you to perform this action.

Click on “Restart Now” to restart your computer (see [Figure 4-12](#)).

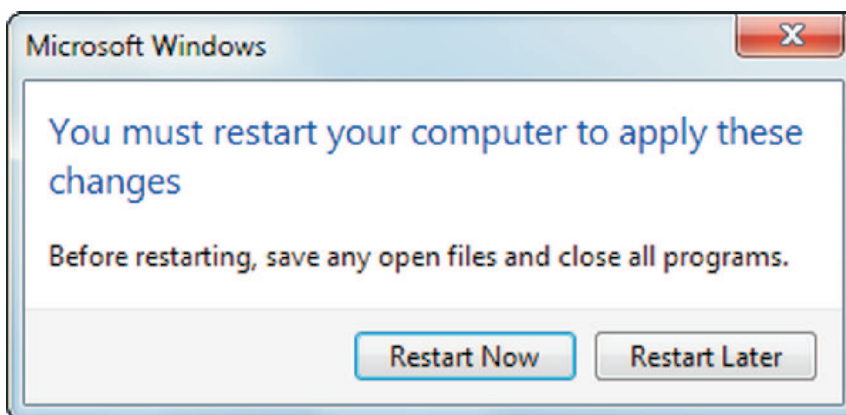


Figure 4-12. Operating System Asking for a Reboot

4.7 Erasing FW in the SPI Flash

In case you want to erase the content of the SPI Flash, it is just a matter of clicking on the “Erase Flash” button in order to issue the erase flash command on your device.

There are two different ways to erase the SPI Flash, depending of the FW already present:

1. HID FW already burned in the SPI Flash

- In this case, you must select the HID interface instance of your choice that is shown in the compatible device list and then click on the “Erase Flash” button (see [Figure 4-13](#)).



Figure 4-13. Erasing SPI Flash With HID FW Already Burned

- Click on “OK” after the erase flash command succeeded (see [Figure 4-14](#)).

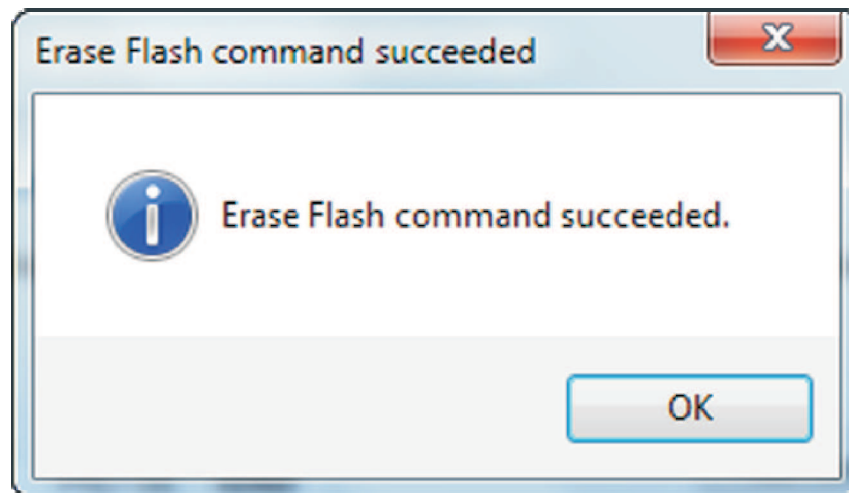


Figure 4-14. Erase Flash Command Succeeded

- After the SPI Flash has been erased, the TUSB9260/TUSB9261 board will re-enumerate with its Boot Loader instance (see [Figure 2-7](#)).
- ### 2. Mass storage only FW already burned in the SPI Flash
- When having a firmware without an HID instance, the “Erase Flash” button can also be accessed when selecting the “USB Mass Storage” instance of our device under the “USB Removable Disk Drives” category.
- From a user perspective, this method is exactly the same we used with the HID instance of our device, that is simply select the Mass Storage device instance and click on the “Erase Flash Button” (see [Figure 4-15](#)). This time the GUI will send the erase flash command through TI (vendor) specific SCSI commands.



Figure 4-15. Erasing the Device's Flash from USB Mass Storage Instance

- Click on “OK” after the erase flash command succeeds (see [Figure 4-14](#)).
- After the SPI Flash has been erased, the TUSB9260/TUSB9261 board will re-enumerate with its Boot Loader instance.

NOTE: Only firmware version 0.95 and above support the TI (vendor) specific SCSI commands required by this function. Please make sure you are using the appropriate FW version.

4.8 Exporting the EEPROM Data to a File

The TUSB926x Flash Burner GUI is also helpful whenever a user wants to burn the EEPROM data using a method other than the FlashBurner GUI as it can export all the required data such as the USB descriptors, checksums and firmware in the appropriate format (as shown in [Figure 3-1](#)).

In order to export the formatted EEPROM image to a file please follow the steps described in [Section 4.3](#) and [Section 4.5](#) and click on the export button as shown below.

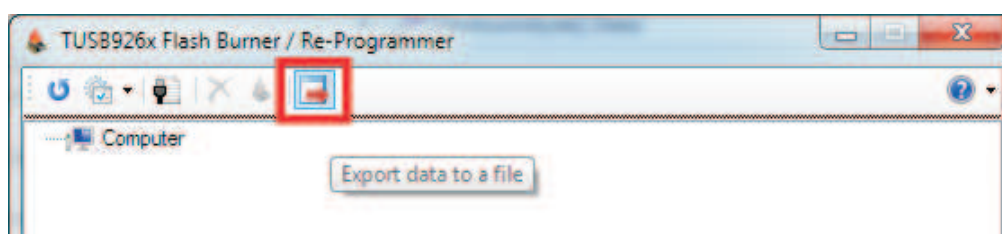


Figure 4-16. Export Button

When prompted, select the folder where you want to store your EEPROM image and the desired name and format. Finally, click the “Save” button as shown below.

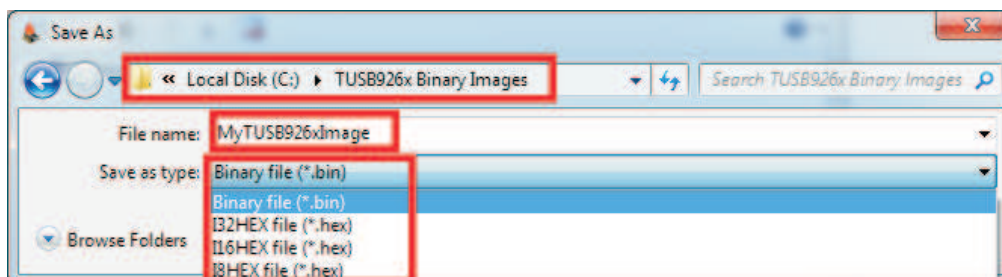


Figure 4-17. Saving the EEPROM

Troubleshooting

During the TUSB9260/TUSB9261 EVM recognition or enumeration process, there might be some troubles that won't let you run the Flash Burner software in an adequate way. All instances should be present in order to burn or reprogram the SPI Flash.

The following are the most common cases that may show up when using the Flash Burner software:

5.1 5.1 Re-Installing Flash Burner Driver Instance Manually

Although it's not necessary, since the Flash Burner GUI will automatically install the driver when required, in this section we will review how to manually install the Flash Burner driver in your system.

In case the TUSB9260/TUSB9261 EVM board comes-up with a "TUSB9260/TUSB9261 Boot Loader" yellow bang instance (see [Figure 5-1](#)), the following steps can be implemented to perform a manual driver installation:

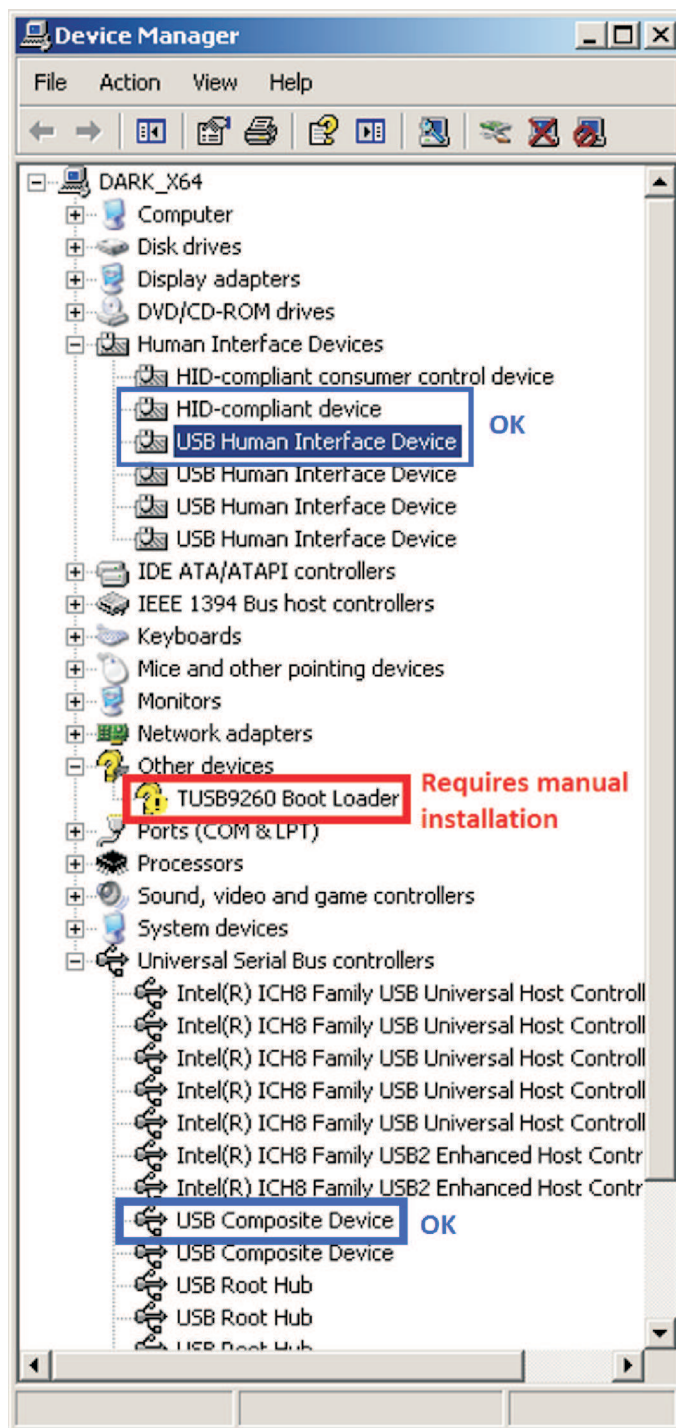


Figure 5-1. TUSB9260/TUSB9261 Boot Loader Instance Not Properly Recognized

Right-click on the “TUSB9260/TUSB9261 Boot Loader” instance and select the “Update Driver...” option (see [Figure 5-2](#)).

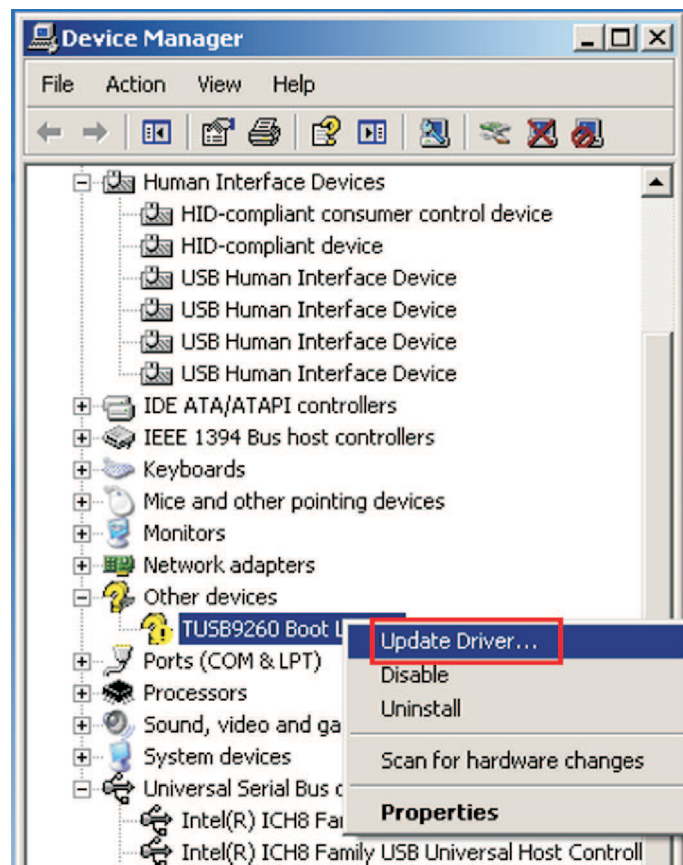


Figure 5-2. Manual Installation for a TUSB9260/TUSB9261 HW Instance

After some seconds, a “Hardware Update Wizard” window will show up. Select “No, not this time” and click on “Next” (see [Figure 5-3](#)).

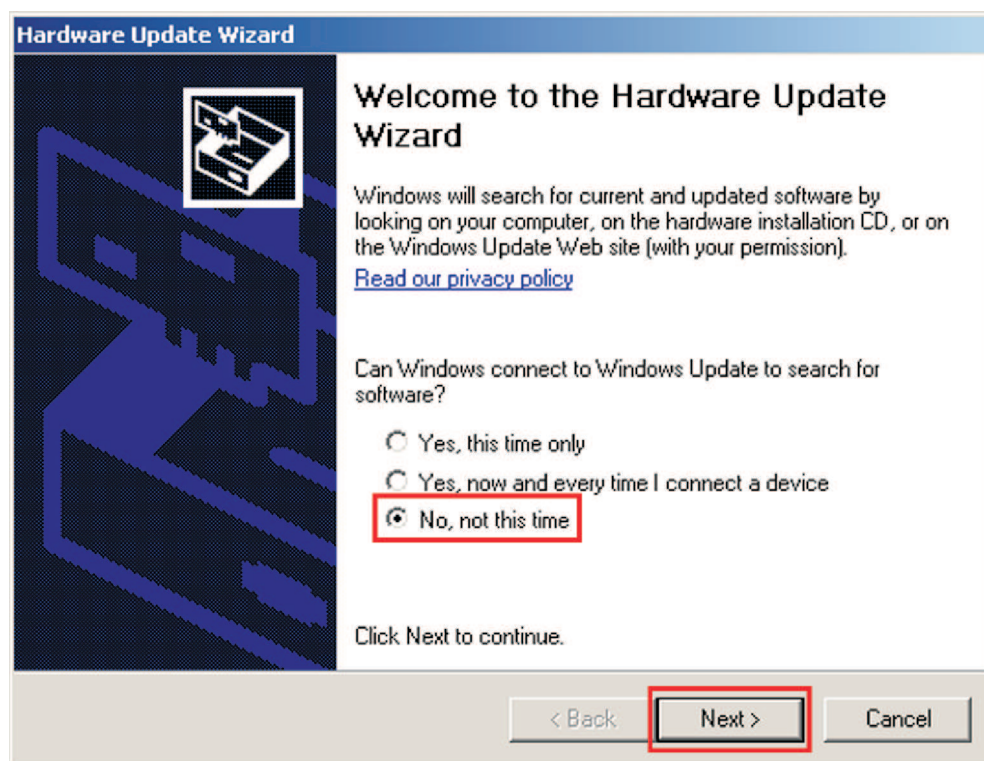


Figure 5-3. Manual Installation Wizard to Avoid Windows Update Online

Since the Flash Burner installer has already copied the proper drivers to the system, you can now select “Install the software automatically...” and click on “Next” to start searching for the proper drivers for such HW instance. Wait until the yellow bang disappears and the instance is proper enumerated under “USB controllers” category (see [Figure 5-4](#)).

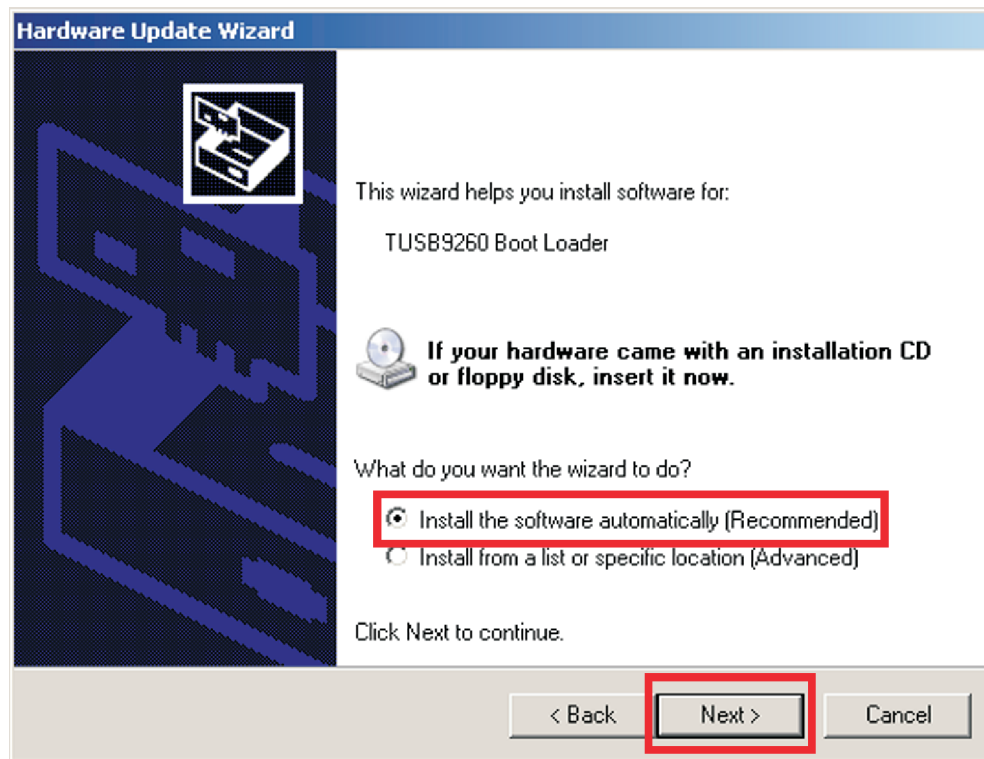


Figure 5-4. Manual Installation Wizard to Perform an Automatic Search

5.2 TUSB9260/TUSB9261 Not Detected Due to a Corrupted FW in the SPI Flash

In case your TUSB9260/TUSB9261 EVM is not being detected by the Flash burner GUI app, it might be possible that the SPI Flash device was not properly programmed or its FW image is corrupted. In order to get your TUSB9260/TUSB9261 EVM detected, you need to let it enumerate using the bootloader embedded in the TUSB9260/TUSB9261 device. To do so, please perform the following steps:

1. Close the Flash Burner GUI application.
2. Unplug your TUSB9260/TUSB9261 EVM.
3. Remove the “SPI Enable” jumper.
4. Plug your TUSB9260/TUSB9261 EVM and wait until it’s properly enumerated and detected.

After the TUSB9260/TUSB9261 bootloader is enumerated, follow these additional steps to properly re-program the SPI Flash:

1. Open the Flash Burner GUI application.
2. Put the “SPI Enable” jumper back on its original position.
3. Follow sections 4.2, 4.3 and 4.4. You will now be able to select the proper device and burn your desired USB descriptors and FW in the SPI flash.

5.3 TUSB9260/TUSB9261 Unrecognized By the Flash Burner GUI and Device Manager

In some cases, after burning a new FW on your device or after installing either the Flash Burner or the Erase Flash driver in one of the device’s instances, Windows can take a considerably long time to properly enumerate the device. While this happens, your device might be listed as “Unknown Device” in Device Manager and the Flash Burner GUI will not show any of your device’s instances.

If that is the case, simply unplug and plug your device back in, and it should be properly enumerated in both the Flash Burner GUI and Device Manager.

5.4 GUI Succeeded, But Device is Not Properly Programmed

When the Flash Burner GUI displays the “Programming Succeeded” message it would indicate that the TUSB926x acknowledged the reception of the FW data, however, there is no implemented procedure to verify that the firmware data was indeed programmed into the SPI flash.

If the device wasn’t properly programmed after the GUIs “Programming Succeeded” message is displayed:

1. Unplug and plug your device back in and wait for the device to be fully enumerated by the Device Manager.
2. Verify your SPI memory supports the following op-codes:

Table 5-1. OP-Codes

OP-CODE DESCRIPTION	OP-CODE DATA
Read JEDEC	0x9F
Write Enable	0x06
Write Disable	0x04
Read Status	0x05
Write Status	0x01
Read Data	0x03
Fast Read	0x0B
Page Program	0x02
Sector Erase	0x20
Block Erase	0xD8
Chip Erase	0xC7
Power Down	0xB9
Release Power Down	0xAB
Manufacturer ID	0x90

3. Verify the connections between the TUSB926x and the SPI Flash and make sure there is activity on those lines while attempting to program the SPI Flash.

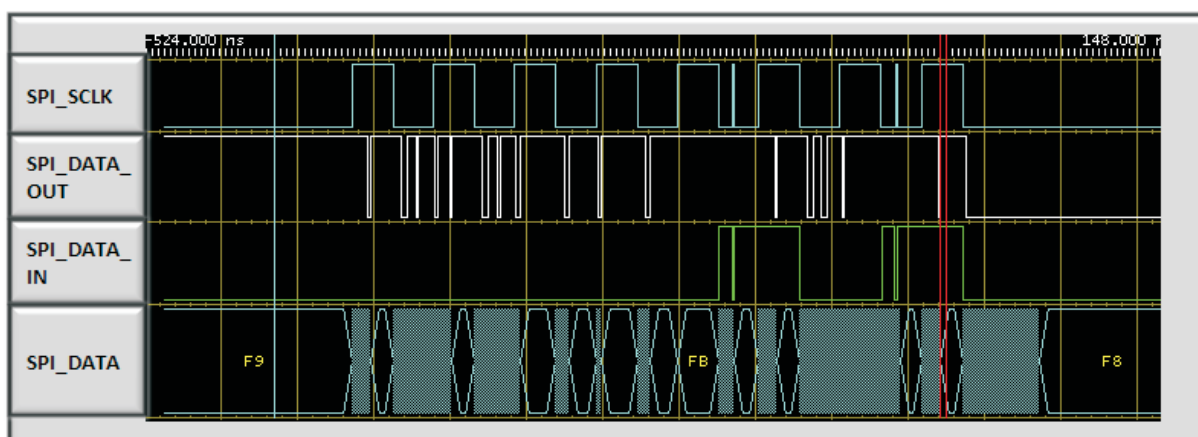


Figure 5-5. Data Transmission During Flash Programming

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products

Audio	www.ti.com/audio
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
RF/IF and ZigBee® Solutions	www.ti.com/lprf

Applications

Communications and Telecom	www.ti.com/communications
Computers and Peripherals	www.ti.com/computers
Consumer Electronics	www.ti.com/consumer-apps
Energy and Lighting	www.ti.com/energy
Industrial	www.ti.com/industrial
Medical	www.ti.com/medical
Security	www.ti.com/security
Space, Avionics and Defense	www.ti.com/space-avionics-defense
Transportation and Automotive	www.ti.com/automotive
Video and Imaging	www.ti.com/video
Wireless	www.ti.com/wireless-apps

TI E2E Community Home Page

e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2011, Texas Instruments Incorporated